P2P Group Management Systems: A Conceptual Analysis

TIMO KOSKELA, OTSO KASSINEN, ERKKI HARJULA, and MIKA YLIANTTILA, University of Oulu

Peer-to-Peer (P2P) networks are becoming eminent platforms for both distributed computing and interpersonal communication. Their role in contemporary multimedia content delivery and communication systems is strong, as witnessed by many popular applications and services. Groups in P2P systems can originate from the relations between humans, or they can be defined with purely technical criteria such as proximity. In this article, we present a conceptual analysis of P2P group management systems. We illustrate how groups are formed using different P2P system architectures, and analyze the advantages and disadvantages of using each P2P system architecture for implementing P2P group management. The evaluation criteria in the analysis are performance, robustness, fairness, suitability for battery-powered devices, scalability, and security. The outcome of the analysis facilitates the selection of an appropriate P2P system architecture for implementing P2P group management in both further research and prototype development.

Categories and Subject Descriptors: A.1 [**Introductory and Survey**]; C.2 [**Computer-Communication Networks**]: C.2.1 Network Architecture and Design—*Network topology,* C.2.4 Distributed Systems—*Distributed databases*; E.1 [**Data Structures**]: Distributed Data Structures; H.3 [**Information Storage and Retrieval**]: H.3.4 Systems and Software—*Distributed systems*

General Terms: Design, Management, Performance

Additional Key Words and Phrases: Peer-to-peer, distributed hash table, overlay network, user community

ACM Reference Format:
Koskela, T., Kassinen, O., Harjula, E., and Ylianttila, M. 2013. P2P group management systems: A conceptual analysis. ACM Comput. Surv. 45, 2, Article 20 (February 2013), 25 pages.
DOI = 10.1145/2431211.2431219 http://doi.acm.org/10.1145/2431211.2431219

1. INTRODUCTION

The need for group management in different kinds of network systems is evident. Groups in networks can exist on many levels [Gupta and Bostrom 2005]. On the one hand, groups can be human-to-human virtual communities, the importance of which is continuously growing as exemplified by the tremendous popularity of social networking services, such as Facebook, Twitter, Google Groups, or Yahoo Groups. On the other hand, groups can be collections of network nodes grouped according to some physical or technical metric such as latency, performance, or proximity.

Financial support from the Finnish Funding Agency for Technology and Innovation (Tekes) within the Future Internet program is gratefully acknowledged. This work has also been supported by the Graduate School in Electronics, Telecommunications and Automation (GETA), Infotech Oulu Graduate School and by scholarship grants from the Foundation of Technology, Nokia Foundation, the Foundation of Walter Ahlström, the Research Foundation of HPY, the Research and Educational Foundation of TeliaSonera, the Foundation of Tauno Tönning, and the Foundation of Riitta and Jorma J. Takanen.
Authors' addresses: T. Koskela (corresponding auhor), O. Kassinen, E. Harjula, and M. Ylianttila, Department of Computer Science and Engineering, University of Oulu, Finland; email: Timo.koskela@ee.oulu.fi.

DOI 10.1145/2431211.2431219 http://doi.acm.org/10.1145/2431211.2431219

The existence of groups is especially emphasized in the case of *Peer-to-Peer* (P2P) networks because they are, inherently, platforms for both interpersonal communication and distributed systems consisting of computing nodes [Androutsellis-Theotokis and Spinellis 2004]. The sharing of resources and information in groups, whether examined from the viewpoint of users or machines, is the core reason why P2P networks exist in the first place [Sun et al. 2006]. Besides the canonical example of multimedia files, the shared resources can include, for instance, different kinds of services, data storage, network bandwidth, or processing capacity [Rodrigues and Druschel 2010]. Whenever the term "group" refers to something other than the entire population of a P2P network, specific solutions are needed in order to manage the P2P group as an entity separate from the rest of the P2P network.

During the last decade, *P2P group management* has emerged as a new area of interest. This is exemplified by the vast number of different kinds of P2P group management systems that have been proposed in the literature for various purposes. To give some examples, groups have been used for enabling real-time group communications [Zhang et al. 2007; Delmastro and Passarella 2005], content and knowledge sharing [Lu and Callan 2003; Wang and Vassileva 2004; Liotta et al. 2005; Li et al. 2007; Anglade et al. 2007a], as well as provisioning of services [Galatopoullos et al. 2008]. In addition, groups have also been used for providing enhanced routing and search efficiency [Garcés-Erice et al. 2003; Mei and Chang 2004; Li and Vuong 2006], trust [Galatopoullos et al. 2008; Ravichandran and Yoon 2006], as well as security and privacy [Arnedo-Moreno and Herrera-Joancomartí 2008; Li et al. 2003] in a P2P system.

Although P2P systems have been widely surveyed by the research community, there exist no comprehensive studies on P2P group management. This article aims to fulfil part of this gap by providing a conceptual analysis of P2P group management systems. We illustrate how groups are formed using different P2P system architectures, and analyze the advantages and disadvantages of each P2P system architecture for the purpose of implementing P2P group management. The evaluation criteria used in the analysis are performance, robustness, fairness, suitability for battery-powered devices, scalability, and security. These properties were selected because they constitute an essential set of challenges faced by P2P group management systems.

Based on the literature survey and the analysis, it is possible for the reader to get a good picture of the overall state of P2P group management research, which facilitates: (1) searching of more detailed information about different P2P group management systems, and (2) selection of an appropriate P2P system architecture for implementing P2P group management in further research and/or development.

The rest of this article is organized as follows: the analysis framework is presented in Section 2, where Section 2.1 provides a classification of P2P systems, Section 2.2 defines the concept of P2P group management, and Section 2.3 overviews the technical challenges in P2P networking. Based on Section 2, Section 3 analyzes the advantages and disadvantages of each different kind of P2P system architecture for implementing P2P group management. Finally, Section 4 provides some guidelines for designing a P2P group management system and concludes the article.

2. ANALYSIS FRAMEWORK

2.1. Classification of Peer-to-Peer Systems

P2P technologies are typically used for implementing overlay networks at the application layer, on top of the underlying (usually IP-based) network topology [Androutsellis-Theotokis and Spinellis 2004]. An overlay network can be defined as a collection of logical links that connect the nodes at the application layer. These logical links consist

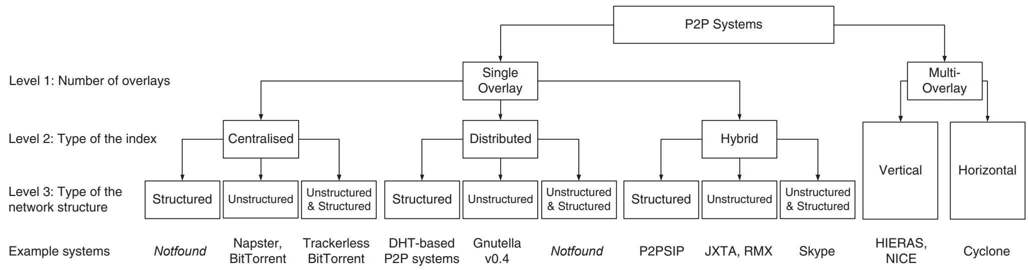

Fig. 1. Classification of P2P systems.

of one or more physical links between the nodes. The topology of a P2P network is independent of that in the lower layers, although information concerning the lower layers (such as network delays and physical proximity) may be utilized in the operation of the P2P network.

In the literature, the concept of P2P networking has been defined in multiple ways. What unites these definitions is that they describe P2P systems as distributed systems that are populated by autonomous and heterogeneous nodes being capable of self-organizing into network topologies without centralized control, for the purpose of sharing resources such as content, CPU cycles, storage, and bandwidth (e.g., Helepovic et al. [2003], Xu et al. [2003], Sakaryan et al. [2004], Androutsellis-Theotokis and Spinellis [2004]). Androutsellis-Theotokis and Spinellis [2004] explain that the diversity of the definitions emerges from the chosen level of broadness attached to the concept. We agree on this, but to be more precise, the broadness itself results from the recent evolution of P2P networking, which has led to the introduction of several architecturally distinct approaches for implementing P2P systems [Harjula et al. 2004]. In order to discuss the details of P2P group management systems, fundamentals of P2P systems must be first understood. Thus, a comprehensive classification of architecturally distinct P2P systems is first provided.

As the basis for our classification of P2P systems (see Figure 1), we have applied relevant parts of the taxonomy of P2P applications presented by Brands and Karagiannis [2009] and added a new level taking into account also the number of overlays that constitute the entire P2P system. The taxonomy introduced by Brands and Karagiannis [2009] was chosen because it is concise and unambiguous, and further, derived from the classifications provided by acknowledged P2P articles such as Androutsellis-Theotokis and Spinellis [2004] and Risson and Moors [2006].

In our classification, we first examine whether a P2P system contains a single overlay network or multiple interconnected overlay networks. As a result, the examined P2P systems are, respectively, classified either as *single-overlay* or *multi-overlay* P2P systems.

The single-overlay P2P systems can be further classified based on their type of index: a P2P system can use *centralized*, *distributed*, or *hybrid* indexing for locating nodes, shared resources, or groups in a P2P network. In centralized indexing, the index of the P2P network is stored in one or multiple centralized servers that are often referred to as trackers [Brands and Karagiannis 2009]. Common examples of P2P systems that use centralized indexing are Napster [2011] and BitTorrent [2011]. In distributed indexing, there is no centralized storage for the index, but the index is distributed among the nodes in a P2P network. For example, P2P systems such as Gnutella v0.4 [Ripeanu 2001] and Freenet [2011] rely on distributed indexing. In hybrid indexing, the responsibility for maintaining the index of a P2P network is distributed to a small subset of nodes called supernodes. Usually, in hybrid P2P systems, the more capable nodes act as supernodes and collectively maintain the index on behalf of the ordinary nodes connected to them [Yang and Garcia-Molina 2003]. In a sense, each supernode acts as a centralized server to a subset of ordinary nodes. In centralized indexing the servers are typically part of the service infrastructure, whereas in hybrid indexing, supernodes are typically users' devices. Examples of P2P systems that use hybrid indexing are Skype [2011], Gnutella v0.6, and FastTrack.

Regardless of the used indexing mechanism, a single-overlay P2P system can be further classified based on its structure, as *unstructured*, *structured*, or a *combination of unstructured and structured*. In practice, the type of overlay structure determines what type of routing algorithm is used [Brands and Karagiannis 2009].

In unstructured P2P systems, the nodes and their shared resources are arranged in an unorganized way [Xu et al. 2003]. In other words, the structure of the network emerges arbitrarily in the course of time. In unstructured P2P systems, resource discovery usually relies on broadcasting, and typically, on a resource-consuming technique called *flooding* [Khambatti et al. 2002a]. In flooding, a search query is propagated from a node to all of its neighbor nodes, then to the neighbors of those nodes, and so on, until the Time-To-Live (TTL) parameter has run to zero. As a result, a search query reaches only a part of the P2P network, and therefore, deterministic resolution of a search query cannot be provided. Some examples of unstructured P2P systems are Napster [2011], Gnutella v0.4, and FastTrack.

In structured P2P systems, a global data structure is maintained [Sun et al. 2006]. The data structure is almost invariably (one of the exceptions is Mercury [Bharambe et al. 2004]) based on algorithms called *Distributed Hash Tables* (DHTs) (e.g., Sun et al. [2006], Xu et al. [2003], Valduriez and Pacitti [2004], Zhao et al. [2002], Artigas et al. [2005]) which map both the nodes and their shared resources into a logical numeric address space. In DHT-based P2P systems, each node and shared resource has its globally unique position (addressed with a key) in the data structure. A key is calculated using a DHT-specific hash function on some string that uniquely describes a node or a shared resource. When the key to a specific node or a shared resource is known, deterministic resolution of a search query can be guaranteed. The most popular DHT algorithms include Chord [Stoica et al. 2001], CAN [Ratnasamy et al. 2001], Pastry [Rowstron and Druschel 2001], Tapestry [Zhao et al. 2004], and Kademlia [Maymounkov and Mazières 2002]. Examples of structured P2P systems include Freenet [2011] and P2PSIP that is based on a protocol called RELOAD [Jennings et al. 2010].

P2P systems that contain both unstructured and structured network topologies are usually based on hybrid indexing. In hybrid P2P systems, the communication between the supernodes is typically structured, whereas the communication between the supernodes and their ordinary nodes is unstructured. The most popular example of a P2P system using the combination of unstructured and structured elements is Skype [2011].

Multi-overlay P2P systems are comprised of multiple interconnected overlay networks that together form a functional entity (e.g., Garcés-Erice et al. [2003], Xu et al. [2003], Zoels et al. [2008], Artigas et al. [2005]). Based on the topological structure and the connection pattern of these multiple overlay networks, multi-overlay P2P systems can be further divided into *vertical* and *horizontal* P2P systems [Artigas et al. 2005]. It should be noted that a multi-overlay P2P system may use different types of index and network structures in the different overlays that together constitute the system.

In vertical P2P systems, multiple overlay networks form a layered structure, where every *layer* is an independent structured (usually DHT-based) P2P overlay network. The communication between the layers is implemented by nominating *gateway nodes* that participate in two vertically adjacent layers [Zoels et al. 2008]. The gateway nodes are responsible for routing of messages between two layers. Examples of vertical P2P systems are HIERAS [Xu et al. 2003] and NICE [Banerjee et al. 2002]. In horizontal

P2P systems, multiple overlay networks, each of them being called a *leaf*, are connected to a single DHT-based P2P network. In addition, there can be connections between the leaf overlays. The responsibility of the single DHT-based P2P network is to optimize the routing in the P2P system and to maintain the conceptual hierarchy of the leaf overlays [Artigas et al. 2005]. It is important to note that there are no specific gateway nodes in horizontal P2P systems, but the leaf overlays are connected by adding some carefully chosen links between the nodes [Zoels et al. 2008]. This is done so that the total number of links per node remains relatively small. Basically, every individual leaf could be based on any type of single-overlay P2P system. Examples of horizontal P2P systems are Cyclone [Artigas et al. 2005] and the framework of semantic communities presented by Li and Vuong [2006].

2.2. Peer-to-Peer Group Management

In the context of P2P networking, the notions of a *group* and a *community* are often used synonymously, even though they possess some conceptual differences [Khambatti et al. 2002b]. Whereas a group is basically a systems term, a community is a social term. From a systems perspective, a group can be defined as *any logically connected subset of nodes that belong to the same P2P overlay network*. These nodes may represent human users, but can also be independent computing entities. Furthermore, it should be noted that the members of a group may or may not be aware of each other. What primarily distinguishes a community from a group are the *social interaction* and the *consciousness of the membership*. Rheingold [2000] defines virtual communities (which are now analogous to communities) as social-motivated webs of personal relationships that share common values and interests, communicated through electronic media such as P2P overlay networks. According to the given definitions, the concept of a group can be seen as a parent class for the concept of a community (i.e., all communities are groups, but not all groups are communities). For simplicity, we use the term “group” to refer to any group, whether the group is a community or not. The social dimension of P2P networking is not examined in detail, because the focus of this article is on the systems side of P2P group management.

Based on our literature survey, *group management* refers to a set of procedures for establishing, configuring, indexing, discovering, joining, leaving, and removing groups, in which the group members are able to share resources and/or to communicate with each other. Consequently, a *P2P group management system* is defined as a P2P system that is able to provide these procedures. Not all P2P group management systems, however, implement all of the listed procedures, but rather an applicable part of them. Moreover, it is important to notice that the execution of the procedures can be either *user-initiated* or *system-initiated*. The term system-initiated refers to procedures that are executed transparently or automatically from the user’s standpoint. For instance, groups might be established based on nodes’ searching behavior [Das et al. 2009] or network proximity [Ratnasamy et al. 2002] even without users’ awareness of the existence of these groups. In Table I, the descriptions of the procedures for P2P group management are provided.

Table II presents an extensive sample of P2P group management systems found in the literature. The identified P2P group management systems are classified according to the type of underlying P2P system.

2.3. Challenges in Peer-to-Peer Networking

The fluent operation of P2P systems is complicated by several technical challenges, which also have an impact on the operation of P2P group management systems. In this section, we summarize the essential technical challenges that affect the operation of P2P systems. The identified challenges involve performance, robustness, fairness,

Table I. Descriptions of the Procedures for P2P Group Management

Procedure	Description
Group establishment	Functionality for creating a new group within a single P2P overlay or as an independent P2P overlay. This procedure might include adjusting configurable parameters of the underlying P2P overlay.
Group configuration	Functionality for modifying group-related descriptive information (e.g. group name) and policies (e.g. membership requirements). This procedure might also include authentication.
Group joining	Functionality for joining an established group. This procedure might also include authentication and evaluation whether membership requirements are fulfilled.
Group leaving	Functionality for leaving an established group. This procedure might also include authentication.
Group removal	Functionality for removing an established group. This procedure might also include authentication.
Group indexing and discovery	Functionality for indexing and discovering groups within a P2P system.
Group member indexing and discovery	Functionality for indexing and discovering members of a group.
Group resource indexing and discovery	Functionality for indexing and discovering resources shared within a group. This procedure might also include access control and encryption.

suitability for battery-powered devices, scalability, and security. In Section 3, these challenges are used for analyzing the advantages and disadvantages of each different P2P system architecture for the purpose of implementing P2P group management.

Performance. This describes the ability to perform the required tasks with an acceptable delay. Suitable metrics for P2P performance often include routing delay or hop count. In P2P networks, end-to-end routing delay is a consequence of the cumulative delay of the data links and intermediary nodes on the communication path [Li et al. 2004]. As the lookup algorithm and the network topology determine the communication path between two previously unfamiliar nodes, they are in key position in defining the routing performance [Hautakorpi and Camarillo 2007]. Lookup hop count denotes through how many logical links the lookup request has to be sent during the lookup phase. Hence, lookup hop count is used as the default routing performance metric in P2P networking.

The importance of performance is especially highlighted in procedures that are executed frequently; group resource indexing and discovery can be an example of a procedure that is invoked frequently or carried out continuously. Performance is also reflected in those group management procedures that are potentially heavy to perform. Depending on the underlying technology, group establishment or group joining can, for example, be such procedures, if extensive messaging is needed in order to create the required bindings between the nodes.

Robustness. Robustness refers to the ability to adapt to changes in the network environment. Dynamicity in a P2P network is caused by both the churn (constant joining and leaving of nodes) and the failures of nodes and network entities. As a result of churn and failures, the message routing infrastructure is effectively breaking, and thus, the messages may not reach their destination. This has a strong negative effect on the success ratio of P2P operations that is commonly used as a metric for evaluating robustness of a P2P system. In the worst-case scenario, churn can incur partitioning of a P2P network into multiple disjoint smaller networks. Partitioning severely affects the functioning of the network, depriving the users of some or all of the network-provided services [Leonard et al. 2008; Yao et al. 2009]. Even if the network does not break into disconnected parts, churn may in many cases affect the performance, lead to the loss of

Table II. Classification of P2P Group Management Systems

	Single-overlay									Multi-overlay	
	Centralised indexing			Distributed indexing			Hybrid indexing				
Publication	Unstr.	Struct.	U. & S.	Unstr.	Struct.	U. & S.	Unstr.	Struct.	U. & S.	Vert.	Hor.
[Agostini and Moro 2004]			x								
[Anglade et al. 2007a]			x								
[Anglade et al. 2007b]			x								
[Artigas et al. 2005]											x
[Banerjee et al. 2002]										x	
[Biswas and Vidyasankar 2008]							x				
[BitTorrent 2011]	x*	x*									
[Bloehdorn et al. 2005]	x										
[Bottazzi et al. 2008]							x				
[Buyukkaya et al. 2008]										x	
[Castano and Montanelli 2005]			x								
[Castro et al. 2002]					x						
[Castro et al. 2003]					x						
[Chawathe et al. 2000]							x				
[Das et al. 2009]			x								
[Delmastro et al. 2005]					x						
[Freedman and Mazières 2003]							x				
[Galatopoullos et al. 2008]											x
[Ganesan et al. 2004]										x	
[Garcés-Erice et al. 2003]										x	
[Gu and Wei 2006]				x						x	
[Harvey et al. 2003]								x			
[Jennings et al. 2010]				x							
[Khambatti et al. 2002b]				x							
[Khambatti et al. 2004]				x							
[Kobayashi et al. 2005]										x	
[Koskela et al. 2010]											
[Krishnamurthy et al. 2001]	x			x							
[Kwok and Gao 2003]				x							
[Lei and Fu 2009]				x							
[Li et al. 2003]							x				
[Li and Vuong 2005]											x
[Li and Vuong 2006]											x
[Li et al. 2007]				x							
[Li et al. 2008]							x				
[Liotta et al. 2005]							x				
[Liu et al. 2006]				x							
[Lu and Callon 2003]							x				
[Mei and Chang 2004]							x				
[Montanelli 2006]				x							
[Morávek and Jelinek 2004]	x										
[Ou et al. 2010]										x	
[Ratnasamy et al. 2002]				x*	x*						
[Ravichandran and Yoon 2006]							x				
[Shan and Shriram 2006]							x				

Continued

Table II. Continued

	Single-overlay									Multi-overlay	
	Centralised indexing			Distributed indexing			Hybrid indexing				
Publication	Unstr.	Struct.	U. & S.	Unstr.	Struct.	U. & S.	Unstr.	Struct.	U. & S.	Vert.	Hor.
[Skype 2011]									x		
[Sripanidkulchai et al. 2003]				x							
[Sun et al. 2006]							x				
[Tham et al. 2004]							x				
[Wang and Vassileva 2004]							x				
[Xu et al. 2003]										x	
[Xue et al. 2004]				x							
[Yamamoto et al. 2008]							x				
[Yu and Li 2008]							x				
[Zhang et al. 2007]				x							
[Zhao et al. 2002]										x	
[Zhuang et al. 2001]					x						

*There are multiple versions of a specific P2P system available or alternative implementation approaches.

data resources, or at least cause extra network maintenance messaging [Leonard et al. 2007, 2008]. Churn tolerance is very important, especially, in P2P systems populated with mobile nodes that tend to participate in the network in a transient manner.

A low level of robustness can lead to problems in any of the procedures for P2P group management. When the structure of the network is prone to break, basic procedures such as group establishment or group joining may fail, since the nodes do not have a reliable connection with each other. Churn and failures may take down nodes that hold important data resources, thus hindering resource-related procedures. Examples of such procedures include all group-related indexing and discovery, as well as group configuration in cases where the implementation of this procedure is based on published data resources.

Fairness. This is the guarantee that a node does not need to contribute more than its fair share of the load in the system. A P2P system is defined to be fair when the service and maintenance load inflicted on a node is shared in a manner that does not let some nodes free-ride at the expense of other nodes [Fan et al. 2009]. If fairness is observable by the users, it has the effect of incentivizing them to participate as the feeling of fairness removes fears of being abused. The used fairness metrics depend on the requirements of a P2P application. The metrics may include, for example, the number of objects a node is responsible for, the number of incoming service requests per node, the total number of messages processed per node, the amount of data transferred to or from a node, a node's CPU load, a node's storage consumption, and a node's energy consumption [Fan et al. 2009]. Fairness of a P2P system can be enhanced, for instance, with load balancing techniques such as those presented in Rao et al. [2003].

In the context of P2P group management, fairness is mainly examined from two viewpoints: fairness between nodes within a group and fairness between nodes that belong to different groups. If groups are implemented within a single overlay, the load generated in one group may also affect the load of the nodes in other groups. In this case, inter-group fairness should be considered especially in all group-related indexing and discovery, because in those procedures the host of a resource may be highly burdened if the resource is, for example, very popular or very large. If, however, each group is implemented as an independent overlay, the load generated in one group has no or little effect on the load of the nodes in other groups.

Suitability for battery-powered devices. As the popularity of mobile computing increases, the problem areas specific to battery-powered terminals become more and more important. Although battery technologies have improved over the time, energy requirements of mobile devices have increased faster [Gupta and Mohapatra 2007; Ravi et al. 2008]. To reverse this development, improving energy conservation methods is essential. For a P2P system, suitability for battery-powered devices is often synonymous with messaging patterns that do not drain the battery too quickly. This is especially challenging in mobile networks, since usually mobile devices are behind a Network Address Translation (NAT) and/or a Firewall (FW) that need to be traversed, increasing the amount of mobile network traffic [Haverinen et al. 2007]. Messaging optimizations may involve mobile-specific modifications to the P2P protocol [Kelényi and Nurminen 2008] or it might be a general property of the P2P system [Jennings et al. 2010]. In addition, ensuring fair load distribution between mobile and fixed devices helps further improve energy efficiency. This can be achieved, for instance, using resource-aware load balancing, or by demoting mobile nodes to a less demanding role in a P2P system, absolving them from the responsibility to contribute to the network management [Nurminen and Nöyränen 2008; Jennings et al. 2010]. The metrics for evaluating the suitability of a networking technology for battery-powered devices include the energy consumption and battery life of a device.

In the context of P2P group management, suitability for battery-powered devices is mainly affected by the devices' role in the maintenance of the P2P network and the indexing of groups, group members, and group resources. The importance of energy consumption is emphasized in group management procedures that are either frequently executed or require intensive signalling, since the energy consumption of mobile networking is especially sensitive to signalling load. The information whether a device is battery powered or not can be taken into account already in group establishment, when their role in the group can be set accordingly.

Scalability. Scalability is the ability to accommodate an increasing number of nodes in a P2P system without severely degrading the performance of the system [Androutsellis-Theotokis and Spinellis 2004]. Scalability is one of the most important evaluation criteria for any distributed networking technology that is intended for widespread deployment. Scalability is an especially important design goal in a P2P system, where an individual node does not have global knowledge of the network state, and the lifetime and capabilities of nodes vary substantially. The increase of hop count in relation to network size is a typical metric for evaluating the scalability of a P2P system. Service discovery is identified, in addition to actual resource searches, as a functionality where scalability is of importance [Ahmed and Boutaba 2011]. Nondeterministic routing and churn-intolerant overlay maintenance are usually identified as factors that can lead to poor scalability.

As practically all group management procedures are based on the sending of messages between nodes using the P2P overlay network, good scalability in the message routing functionality supports all the procedures. Moreover, scalability has special influence on group establishment and group joining procedures. In these procedures, scalability affects how many groups can exist simultaneously, and how many members can participate in each of these groups without any significant decrease in the performance of the system.

Security. This is the level of information security provided in a P2P system. As the operation of P2P systems relies on a large number of potentially hostile nodes, security measures may require special attention in the system. Security properties of P2P systems can be examined from two different viewpoints. One approach is to list the desirable properties of a secure P2P system [Androutsellis-Theotokis and Spinellis 2004]. The other approach is to list the security problems or possible attacks against a P2P system that could violate the security properties of the P2P system [Palomar et al. 2006]. Androutsellis-Theotokis and Spinellis [2004] classify security properties into integrity and authenticity, privacy and confidentiality, and availability and persistence, whereas Palomar et al. [2006] classify security properties into man-in-the-middle, Sybil protection, and node anonymity, to mention a few examples. Security and privacy issues specific to the emerging P2P streaming systems have also been addressed in Gheorghe et al. [2011].

Security is essential in all group management procedures. Group joining is an obvious example: sometimes only authenticated nodes are permitted to become members in a group. Similarly, security enforcement may be needed against, for example, unauthorized group removal or group leaving. Confidentiality, integrity, and availability are important for any sensitive information handled in the context of procedures such as group member indexing and discovery or group resource indexing and discovery. This is especially evident when groups are implemented within a single overlay. Security policies can be part of the settings managed using group configuration procedures.

3. ANALYSIS OF DIFFERENT P2P SYSTEMS FOR IMPLEMENTING GROUP MANAGEMENT

In this section, we present how groups are typically established using different P2P system architectures. In addition, we also analyze the advantages and disadvantages of each different P2P system architecture for the purpose of implementing P2P group management. Based on Section 2.3, the evaluation criteria used in the analysis are performance, robustness, fairness, suitability for battery-powered devices, scalability, and security.

3.1. Single Overlay

In single-overlay P2P systems, one or multiple groups are established within a single overlay network. Depending on the particular realization, the groups can be overlapping. Based on Section 2.2, single-overlay P2P systems are classified into the following three categories based on their index: centralized, distributed, and hybrid.

3.1.1. Centralized Indexing. In P2P group management systems that rely on centralized indexing, at the very least, groups are indexed and discovered using centralized servers. The rest of the group management procedures can also be implemented using centralized servers, but alternatively, they can be implemented using P2P overlay networks. Figure 2 illustrates how groups are formed in P2P systems that use centralized indexing.

Real-life examples of unstructured P2P systems that use centralized indexing are Napster [2011] (Figure 2(a1)) and the original version of BitTorrent [2011] (Figure 2(a2)). In Napster, all nodes connect to a centralized server that is responsible for all indexing and discovery in the system. Only the actual file transfers are conducted in a P2P manner. Napster does not inherently support groups, but in Napster-like P2P systems, all group management procedures could be implemented on the centralized servers. In BitTorrent, a group exists only for the purpose of distributing a specific resource. Thus, indexing and discovery for both groups and group resources are essentially the same group management procedure. In the original version of BitTorrent, group resource indexing and discovery is collaboratively managed by two types of centralized servers: *index servers* and *trackers*. Information about each group resource is stored on index servers in a file format called *torrent*. A torrent is a small file that contains metadata about the data resource and the contact information of the tracker responsible for coordinating the distribution of the data resource. Under the coordination of a tracker, nodes are able to establish connections with each other and form a file sharing group that is typically referred to as a *swarm*. Trackers implement the indexing and discovery of group members (swarm members)

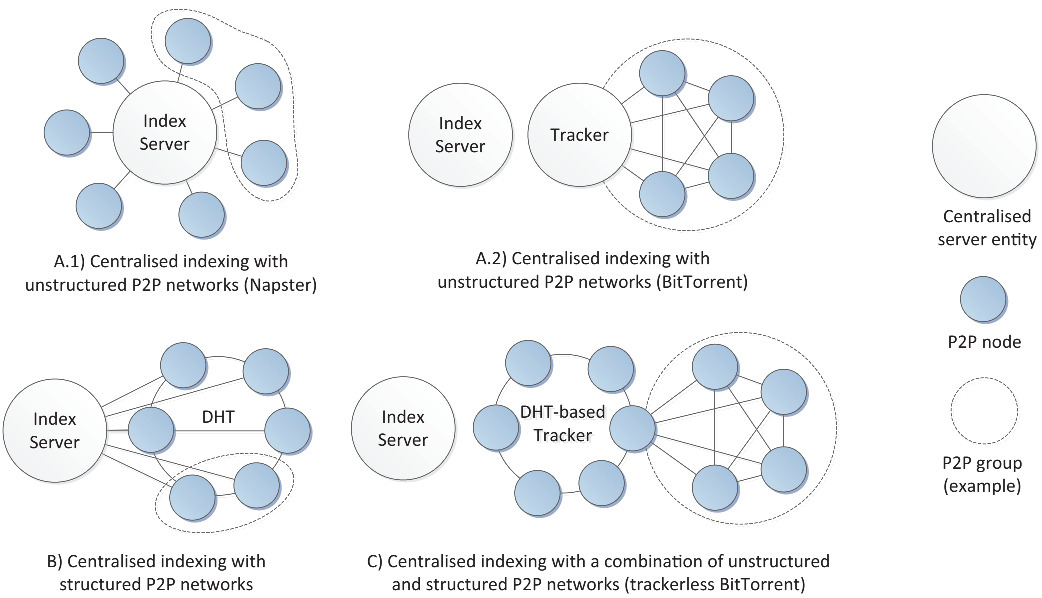

Fig. 2. Groups in P2P systems that use centralized indexing.

and also partly indexing and discovery of group resources, since they keep track of the nodes where a specific piece of the data resource can be found. In addition, trackers are responsible for conducting group establishment, group configuration, group joining, group leaving, and group removal procedures. In the context of BitTorrent, group configuration usually stands for policies that regulate the priorities in the file distribution, for instance, whether the rarest pieces of a data resource are downloaded first or the ones that can be acquired most swiftly. In real-life application, it is rather typical that the same server entity acts as both an index server and a tracker.

While conducting our literature survey, we did not find any examples of structured P2P systems that use centralized indexing. However, an illustrative example could be a DHT-based P2P system, where all group management procedures are implemented by a centralized server (Figure 2(b)). Thus, all group-related indexing and discovery would be conducted using the centralized server. In practice, a group would exist as a list of the DHT-based identifiers of nodes and their shared resources.

The trackerless version of BitTorrent [2011] (Figure 2(c)) provides an example of a P2P system that uses centralized indexing and a combination of unstructured and structured P2P overlay networks. In the trackerless version of BitTorrent, the centralized trackers have been replaced by a DHT-based P2P overlay network. Hence, instead of an address of a centralized tracker, a torrent file contains an *infohash* that is an address in the DHT-based P2P overlay network, where the list of members of a specific file sharing group is stored. The node that is responsible for maintaining the member list assumes the role of a tracker. In the trackerless version of BitTorrent, the only centralized entity is the index server.

Performance of P2P group management systems that use centralized indexing is usually very good, because the discovery of groups, group members, and group resources requires only O(1) hops resulting in minimal routing delays. Performance tends to be the better the more group management procedures are implemented on centralized servers instead of P2P overlay networks. For instance, the performance of the trackerless version of BitTorrent is mainly dependent on the performance of its DHT-based P2P overlay network, where the required number of hops is O(*log n*).

Robustness of P2P group management systems that use centralized indexing must be analyzed from two different angles. On one hand, centralized servers tolerate churn very well, since there is no specific network topology of nodes to be maintained. On the other hand, centralized servers form a single point of failure. As a result of a server failure, the operation of a group could be interrupted, or in the worst case, all information about the group could be lost.

Fairness of P2P group management systems that use centralized indexing depends on how indexing and discovery of groups, group members, and group resources as well as the exchange of group resources are implemented. Use of centralized servers for indexing and discovery provides a good level of fairness from a user's standpoint, since the load on a node is mainly dependent on the extent of the node's activity. However, for example in Napster-like P2P systems, nodes with more popular resources might be under heavier load. In BitTorrent, the fairness in the file sharing groups can be further increased by using policies that guide nodes to send data to those nodes that send data back to them. When using DHT-based P2P overlay networks for implementing group-related indexing and discovery, the load is usually well distributed, at least as long as the nodes are relatively homogeneous in terms of capacity.

Use of centralized indexing increases the suitability of a P2P group management system for battery-powered devices. This is mainly a result from the fact that with centralized indexing battery-powered devices are not burdened by group-related indexing and discovery procedures. In Napster-like systems, a battery-powered device needs to establish connections only with the centralized server, and additionally with some other nodes when exchanging data resources. The small number of required connections also alleviates the burden caused by the NAT and FW traversal. In BitTorrent, the challenge caused by the NAT and FW traversal is dependent on the size of the file sharing group and the used policies for file distribution.

Scalability of P2P group management systems that use centralized indexing is usually poor, since the centralized servers have their own capacity limits that can only be increased by acquiring new hardware components. This increases not only the acquisition costs, but also administrative and maintenance costs. Due to the restricted capacity, the number of groups and the size of groups may be limited. Scalability has been improved, for instance, in the trackerless version of BitTorrent by replacing the centralized trackers with a DHT-based P2P overlay network.

Security properties of P2P group management procedures that rely on centralized server entities are fairly straightforward to manage. For example, ill-behaving group members can be easily eliminated from the index by the centralized server. In BitTorrent, the integrity of each downloaded data piece can be verified using its SHA-1 hash code stored in the torrent file. However, there are no guarantees that the contents of a torrent file are what was expected or that a tracker, whether a server or a P2P node, is not hostile. To alleviate this issue, the contents of a torrent file are typically reviewed by several users, which helps improving the security of using BitTorrent.

3.1.2. Distributed Indexing. In P2P group management systems that rely on distributed indexing, there exist two predominant approaches for implementing group-related indexing and discovery. In the first approach, all group-related indexing and discovery is implemented in a distributed manner using tree-like structures on top of a P2P protocol. In the second approach, indexing and discovery of group resources is implemented in a distributed manner, but indexing and discovery for groups and their members is merely defined locally by each individual node. Figure 3 illustrates how groups are formed in the P2P systems that use distributed indexing.

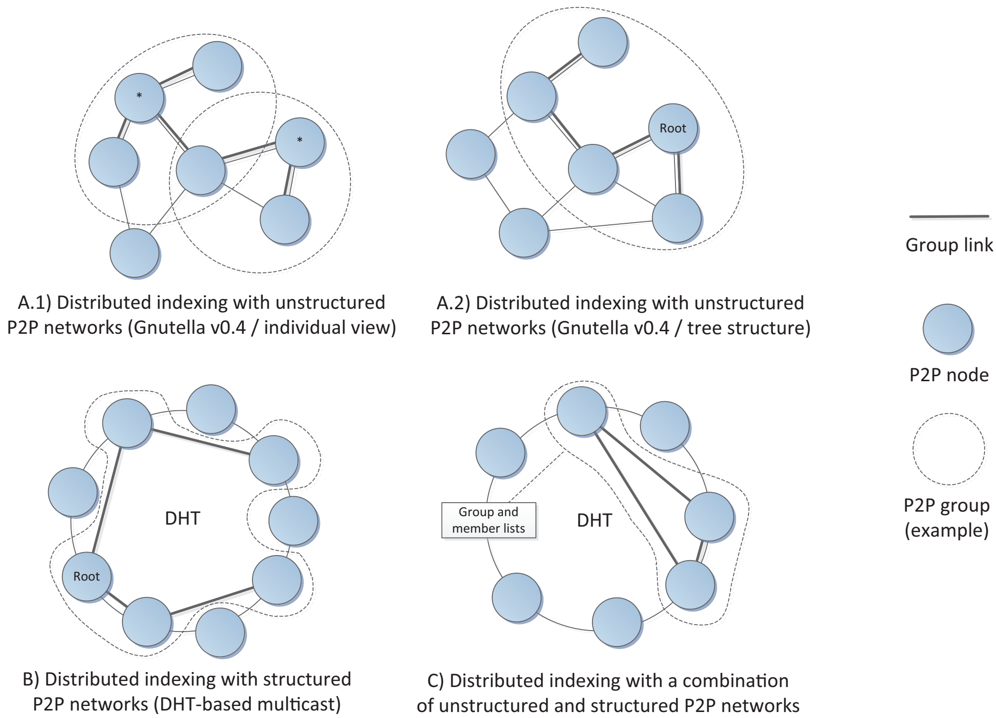

Fig. 3. Groups in P2P systems that use distributed indexing.

An example of an unstructured P2P system that uses distributed indexing is Gnutella v0.4. In Gnutella v0.4, all nodes establish connections with their neighbors forming a random network topology of interconnected nodes. Gnutella does not inherently support groups, but many Gnutella-like P2P systems have been extended with group management procedures [Castano and Montanelli 2005; Das et al. 2009; Khambatti et al. 2004; Li et al. 2007]. There exist two prominent approaches for implementing group management in Gnutella-like P2P systems. The first approach is to establish groups from the perspective of each individual node (Figure 3(a1)). For instance, in Das et al. [2009], groups are formed in a lightweight manner by adding *community edges* between nodes. A community edge is a link that brings two nodes at a one-hop distance from each other. As a group is always established from the standpoint of an individual node, it is unlikely that all nodes in a group (seen by a particular node) have a direct connection also with each other. A new community edge can be added, for example, when a node's search query has been successfully responded to by another node. As multiple groups can be greatly overlapping, all the community edges in the P2P system form a graph structure. The second approach is to establish an additional structure (usually a tree) on top of an unstructured P2P system (Figure 3(a2)). For instance, in Castano and Montanelli [2005], the founder of a group defines the membership requirements and sends an invitation to its neighboring nodes. The invitation message contains a TTL parameter, which defines the maximum number of hops the invitation is propagated in the P2P network. Each node who matches the membership requirements joins the group by contacting the founder. The nodes in a group form a tree, where the founder acts as the root. In practice, groups in Gnutella-like P2P systems emerge in time rather than are established in a controlled manner [Castano and Montanelli 2005]. In this context, the concept of a group remains rather loose, because the groups may be strongly overlapping, and in addition, the members of the same group might not be aware of each other [Agostini and Moro 2004]. It is also common that the execution of the group management procedures is solely system-initiated.

Examples of structured P2P systems that rely on distributed indexing are basically all DHT-based P2P systems that do not contain node hierarchy. In these P2P systems, all group-related indexing and discovery is implemented in a distributed manner, and the groups are usually established using a separate multicast protocol on top of the DHT algorithm (Figure 3 (b)). The use of multicast structures, such as trees, is enabled by the deterministic resolution of queries. For instance, in Castro et al. [2002], an application-level multicast infrastructure called Scribe is used on top of Pastry. In Scribe, a group is established by creating a special DHT identifier called *topicId* that is set as the root of a multicast tree. The multicast tree is formed as a union of DHT routes from each group member to the topicId. A node that is responsible for storing a group's topicId acts as a rendezvous point for that particular group. A rendezvous point is used for joining and leaving a group and disseminating multicast messages to the group members. Other examples of multicast infrastructures are Bayeux [Zhuang et al. 2001] and SplitStream [Castro et al. 2003].

While conducting our literature survey, we did not find any examples of P2P systems that uses distributed indexing and a combination of unstructured and structured P2P overlay networks. However, an illustrative example could be a system where indexing and discovery for both groups and their members is implemented using a DHT-based P2P network (Figure 3(c)). This could be done, for instance, by publishing a list of groups in the DHT-based P2P network using a generally known DHT identifier. Furthermore, the member list of a specific group can be discovered using the group's DHT identifier, known from the group list. A node can join a group by adding itself to the member list and by establishing direct connections with the members of a group. The groups would thus exist as unstructured P2P networks, where indexing and discovery of group resources take place. This example resembles a horizontal multi-overlay P2P system, but is distinguished from it by the fact that now the DHT-based overlay is used for indexing and discovery of groups and their members, not for managing the conceptual hierarchy of the P2P system or any routing optimizations.

Performance of P2P group management systems that use distributed indexing depends essentially on the type of the used P2P overlay network. In general, the performance of unstructured P2P systems is hard to guarantee because of their unorganized messaging patterns [Zhang et al. 2007]. However, especially when group members are connected with direct links, they are reachable with only one hop. This improves the performance of Gnutella-like P2P systems from the viewpoint of implementing group management, but only if the network size stays moderate. The DHT-based P2P systems have potential for very good performance in P2P group management because of the low and predictable hop count. This enables fast routing in multicast infrastructures implemented on top of DHT-based P2P networks.

When groups in Gnutella-like P2P systems are established as a collection of links between the pairs of nodes, no group-related information needs to be collectively maintained. This has a significant positive effect on the level of robustness. If the groups are established as trees, the structure of a tree must be repaired every time a node joins or leaves the group. This introduces a challenge to group management regardless of whether unstructured or DHT-based P2P systems are used. It should also be noted that each root node introduces a potential single point of failure. In contrast to unstructured P2P systems, also the underlying DHT structure of DHT-based P2P systems requires continuous management in the presence of churn or node failures [Zhang et al. 2007].

In unstructured P2P networks that rely on distributed indexing, fairness within the groups is at a reasonable level if the groups are not eminently overlapping. If a node belongs to multiple groups or acts as a root node of a tree, this particular node might become constantly burdened by search queries. In addition, it should be noted that also nongroup-related flooding can burden the nodes in a group, and especially the ones that are in a central location in an unstructured network topology. With DHT-based P2P systems, the messages are fairly evenly distributed between the nodes by the DHT algorithm itself. However, multicast infrastructures in DHT-based P2P systems might require group-related message forwarding services from nodes that are not part of the particular group [Castro et al. 2002]. In addition, a root node of a multicast tree gets more than the fair share of the incoming requests, because it manages group joining and leaving procedures as well as the initiation of multicast messages.

Use of distributed indexing might compromise the suitability of a P2P group management system for battery-powered devices. With unstructured P2P systems, the burden from network maintenance messaging is generally low. However, the unorganized messaging patterns might generate lots of network traffic, especially if the groups are not used for strictly restricting the scope of messaging. For instance, a node might first send a query to a node that is a member of its group, but if no answer is found, the query might be flooded to the rest of the P2P network. In DHT-based P2P systems, battery-powered nodes mainly suffer from a large amount of network maintenance messaging if not taken into account in the P2P protocol design [Kelényi and Nurminen 2008]. Finally, it should be noted that NAT traversal is also an issue for all P2P systems that use distributed indexing, because several different nodes need to be able to initiate connections towards battery-powered nodes.

Scalability of P2P group management systems that use distributed indexing is usually good. Unstructured P2P systems themselves, however, are not very scalable [Sripanidkulchai et al. 2003]. This affects the scalability of tree-like structures, but scalability is not a major issue when most of the group management procedures are implemented locally by each individual node. Thus, the performance is not decreased if the average group size stays moderate. The scalability of P2P group management systems that rely on DHT algorithms is usually good, thanks to the very good scalability of the underlying DHTs for large-scale systems [Harvey et al. 2003]. In a typical DHT, the hop count increases logarithmically when the number of nodes increases linearly. Thus, even in relatively large multicast groups, routing of messages is efficient.

Security properties of P2P group management procedures that rely on distributed indexing might require special attention. In unstructured P2P systems, security may need enforcement, since the flooding of packets to a large number of unknown hosts could compromise the confidentiality and integrity of the data. It must be noted that even though groups exist, they may not set any boundaries to the routing of messages [Das et al. 2009]. In contrast to unstructured P2P systems, security properties are somewhat easier to manage in DHT-based P2P systems. This is because of the fact that routing of messages is more controlled, and thus each message is processed only by a small number of nodes. However, in DHT-based P2P systems, security may also need enforcement, because the group-related multicast messages may travel through potentially hostile DHT nodes [Harvey et al. 2003].

3.1.3. Hybrid Indexing. In P2P group management systems that rely on hybrid indexing, all group management procedures are implemented by supernodes. Ordinary nodes are able to discover and join groups by interacting with the supernode they are currently connected to. With hybrid indexing, groups are usually formed by one or multiple supernodes and their connected ordinary nodes. If multiple supernodes constitute a group, each supernode must be able to discover other supernodes in that particular group. In addition, multicast infrastructures can be used for establishing groups when

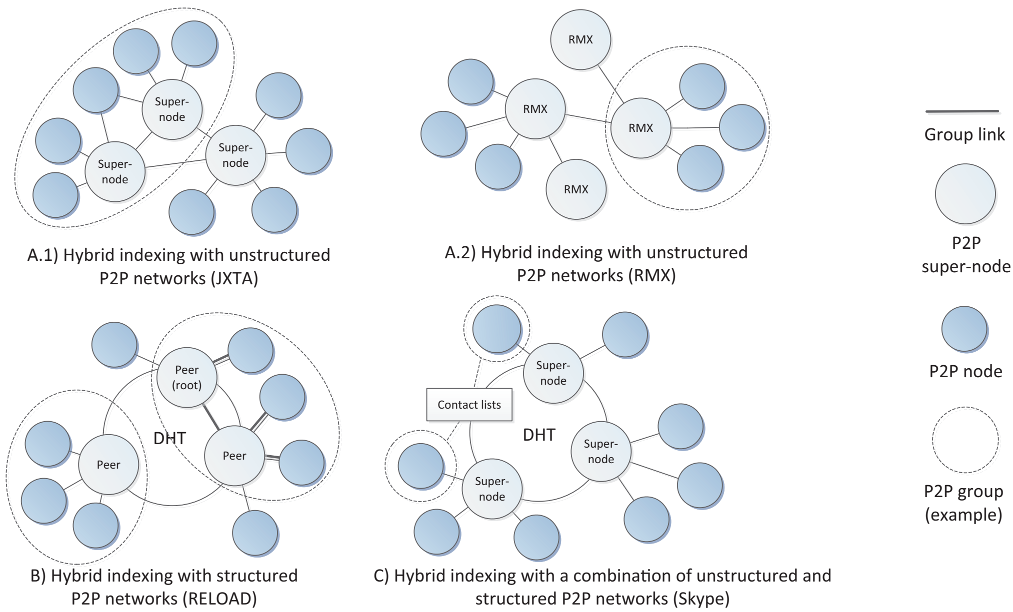

Fig. 4. Groups in P2P systems that use hybrid indexing.

the supernodes are connected using a DHT-based P2P network. Figure 4 illustrates how groups are formed in P2P systems that use hybrid indexing.

The project JXTA [Gong 2001] (Figure 4(a1)) and Reliable Multicast Proxy (RMX) [Chawathe et al. 2000] (Figure 4(b1)) are good examples of unstructured P2P systems that rely on hybrid indexing. The project JXTA is a set of protocols for different P2P functionalities. In JXTA, supernodes can be classified into *rendezvous peers* acting as peer intercommunication hubs and *relay peers* assisting in FW traversal. The groups in JXTA are realized as collections of nodes that share at least one common supernode. To enable propagation of messages among the members of a group, rendezvous peers cooperate with each other and with the ordinary nodes. In JXTA, the creator of a group has freedom to configure the group policies, for instance, by defining membership requirements or even by modifying the default routing scheme used within the group. In RMX, a hybrid multicast infrastructure is established, where homogeneous nodes are grouped together. Each group is managed by an RMX (i.e., supernode) that together form a spanning-tree on the application layer. The communication between the RMXs is implemented using spanning-tree flooding, whereas each RMX uses IP multicast to send data to its group members. In both JXTA and RMX, the supernodes are interconnected using an unstructured P2P overlay network that enables the discovery of remote groups. Differences between the two include, for example, that RMX supernodes are specifically instructed to be placed in strategic locations in the network, while in JXTA nearly any node can assume the role of supernode if needed. Furthermore, in RMX the supernodes may dynamically alter the content of the handled data flows based on application-level knowledge.

An example of a structured P2P system that relies on hybrid indexing is a P2PSIP protocol called RELOAD [Jennings et al. 2010] (Figure 4 (b)). To avoid confusion, it is important to point out that RELOAD uses a different kind of terminology for describing node hierarchy. In RELOAD, a peer refers to a supernode and a client to an ordinary node. Both peers and clients belong to the same DHT-based numerical address space, which is designed so that each peer is responsible for those clients that are near in the address space. RELOAD does not inherently provide group management procedures, but a peer and its connected clients could be considered as a group. Furthermore, a multicast infrastructure could be used for group establishment. With RELOAD, a multicast tree would be naturally constructed so that the branches are formed by the peers and the leafs are the clients.

A real-life example of a P2P system that uses hybrid indexing and a combination of unstructured and structured P2P overlay networks is Skype [2011] (Figure 4 (c)). The network of supernodes is based on a DHT algorithm, but the communication between the supernodes and their ordinary nodes is implemented using some sort of flooding [Brands and Karagiannis 2009]. From a user's viewpoint, a group in Skype appears in the form of a contact list that is maintained by supernodes in a distributed manner. However, from a technical viewpoint, a group in Skype-like P2P systems could also be defined as a collection of supernodes (whether one or multiple) and their connected ordinary nodes, or alternatively, as a multicast infrastructure.

Performance of P2P group management systems that rely on hybrid indexing is good, because interaction between two ordinary nodes (i.e., group members) that share the same supernode requires only two hops. If the two ordinary nodes are connected to different supernodes, only a few additional hops are required, because supernodes belonging to the same group must be able to easily discover each other. Discovery of remote groups is also rather efficient, because only a few hops are needed in the relatively small network of supernodes.

There are two aspects to the robustness of P2P group management systems that use hybrid indexing: (1) if an ordinary node joins or leaves a group, only a minor amount of traffic takes place between a supernode and that specific ordinary node; (2) if there is only a single supernode in a group and it leaves the group, a new supernode needs to be assigned for the group or the whole group ceases to exist [Sun et al. 2006]. In comparison to unstructured P2P systems, a supernode is easier to replace in DHT-based P2P systems, if the group is tied to a certain DHT identifier such as in Scribe [Castro et al. 2002]. In the case of a supernode failure, the nearest supernode in the DHT address space could automatically replace the failed supernode. Alternatively, a new supernode could also be elevated from the group of ordinary nodes.

To guarantee fairness, particularly the choosing of supernodes must be based on agreeable policies. This is because the supernodes often bear the largest burden in a group. These policies may contain rules, for instance, on the required minimum capabilities of a supernode or the maximum number of its connected ordinary nodes [Liotta et al. 2005]. In P2P group management systems that use hybrid indexing, fairness is inherently enhanced by the fact that group-related messaging does not affect the ordinary nodes in the other groups. However, in DHT-based P2P systems, group-related indexing and discovery might require message forwarding and data storage services from supernodes that are not part of the particular group. This might have a negative effect on the fairness if some groups are more active than the others.

Use of hybrid indexing in P2P group management systems has a significant positive effect on suitability for battery-powered devices. By assigning mobile nodes as ordinary nodes instead of supernodes in a group, the burden on these battery-powered devices can be significantly decreased. From the viewpoint of ordinary nodes, NAT traversal is only a minor problem, since they need to maintain the connection only to a single supernode. Mobile nodes can also become supernodes, but then NATs and battery life may become issues regardless of whether unstructured or DHT-based P2P systems are used.

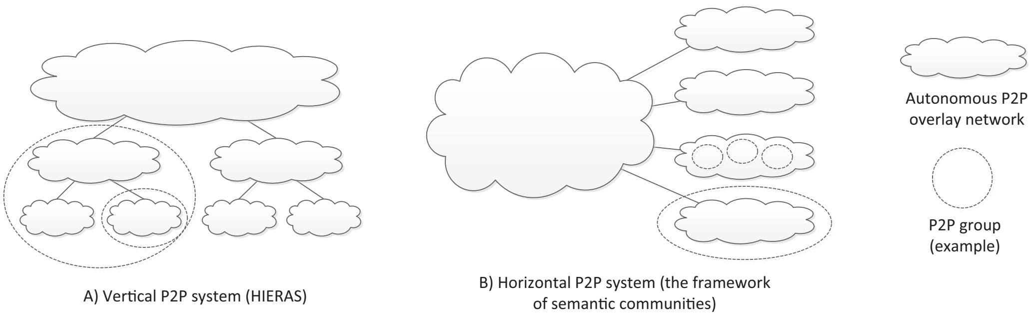

Fig. 5. Groups in multi-overlay P2P systems.

Scalability of P2P group management systems that rely on hybrid indexing is fairly good [Shan and Shriram 2006], because: (1) the size of the P2P network formed by the supernodes is relatively small, allowing for low-cost addition of new groups and a quick discovery of remote groups; and (2) the communication within a group has a very simple messaging pattern. When a group grows large and has only a single supernode, this supernode may become a bottleneck, however. Thus, the implementation of the P2P group management should take into account that several supernodes per group may be needed.

In unstructured P2P systems that use hybrid indexing, security within a group is at a good level, since the scope of messaging is restricted solely to the members of a group; of course, the nodes chosen as the group's supernodes should be trusted [Galatopoullos et al. 2008]. In DHT-based P2P systems, security within a group may need enforcement, because part of the communication may be routed by supernodes that are outside of the immediate group. Moreover, the group resources can be assigned to any supernode in the DHT-based P2P system by the DHT algorithm.

3.2. Multi-Overlay

In multi-overlay P2P systems, one or multiple groups are established using multiple interconnected overlay networks. Usually, one P2P overlay network is used for each group. However, also other kinds of approaches can exist, where either one P2P overlay network contains multiple groups or one group is spread across multiple P2P overlay networks. Based on Section 2.2, multi-overlay P2P systems are classified as either vertical and horizontal according to their topological structure and the connection pattern of the multiple overlay networks.

3.2.1. Vertical. In vertical P2P systems, a group consists of one (or multiple) structured P2P networks. Examples of vertical P2P systems are HIERAS [Xu et al. 2003] (Figure 5(a)) and NICE [Banerjee et al. 2002]. In HIERAS, the entire P2P system is divided into several layers. The highest layer contains all nodes in a single DHT-based overlay, whereas each lower layer consists of multiple DHT-based P2P overlays that correspond to groups. Each node must belong to one group in each layer and be capable of acting as a gateway node between two adjacent layers. In order to minimize the network latencies, topologically adjacent nodes are grouped together in such a way that the smallest network latencies occur in the lowest layer. Most of the routing hops in HIERAS take place in the lower layers before reaching the highest layer, which results in smaller total routing delay. In NICE, nodes are arranged into layers and further into groups based on node proximity. From each group a node that is in a central position to the other group members in terms of distance is nominated as a group leader. The reason for this is to guarantee quick and lightweight joining of new members without burdening the group with extensive messaging. In NICE, all nodes are part of the lowest layer, but each cluster leader also takes part in the next layer above. Finally, the highest layer has only a single group with a single member. The vertical structure of NICE is used for enabling efficient and scalable multicasting on the application layer.

Performance of vertical P2P systems for group management is very good, since the nodes are usually assigned into groups based on performance metrics such as routing latency. The complexity of the system, however, hinders robustness, since churn may have an effect on multiple groups residing in separate P2P overlay networks. Fairness naturally depends on the details of the given system, but is generally good, because most group-related messaging occurs within the boundaries of the established groups. As regards suitability for battery-powered devices, it should be noted that a mobile node may need to participate in multiple P2P overlay networks, and thus take responsibility of maintaining multiple network structures [Xu et al. 2003]. If these P2P overlay networks are small-sized, the burden inflicted on a mobile node might still be lighter than, for instance, in a single large DHT-based P2P overlay network, because maintaining a single large DHT may generate more maintenance load. The NAT traversal is a problem, because several connections are needed towards the mobile nodes. Vertical P2P systems have good scalability, since the routing of messages within and between groups can exploit the routing-related information contained in the structure of the multilayer system, which may be fine-tuned, for example, according to routing latencies or geographical proximity [Harvey et al. 2003]. Security may need enforcement, because establishment of groups is usually not based on trust. Thus, messages within the groups or between different groups may be routed through potentially hostile nodes.

3.2.2. Horizontal. In horizontal P2P systems, groups are established as autonomous P2P networks called leafs that are managed by a common DHT-based P2P network [Artigas et al. 2005]. Examples of horizontal P2P systems are the framework of semantic communities presented by Li and Vuong [2006] (Figure 5 (b)) and Cyclone [Artigas et al. 2005]. In Li and Vuong [2006], a DHT-based P2P overlay is created: (1) to assist nodes organizing into groups, (2) to discover entry nodes to join a specific group, and (3) to forward queries to the correct groups. The DHT-based overlay consists of nodes that have high capabilities and long online time. Instead of the DHT-based P2P overlay, the groups are constructed as unstructured P2P systems in order to enable the use of complex queries. In Cyclone, a tree-like conceptual hierarchy of autonomous DHT-based P2P overlay networks (i.e., groups) is created. Each node is given an identifier that is comprised of two parts: a unique intra-group identity and a group identifier. The established groups in the conceptual hierarchy are not fully connected, but rather use some carefully chosen inter-group links [Zoels et al. 2008].

Performance of horizontal P2P systems for group management is dependent on the type of the P2P system applied within a group overlay. As for robustness, it is important to notice that the groups are autonomous and churn in one of the groups has no effect on the other groups. However, a failure of a node that has some responsibilities for linking groups in the conceptual hierarchy might cause some system-wide problems. Fairness measures are also derivable from the types of P2P networks used in the group overlays, but the nodes that also take part in the common DHT-based P2P network are naturally under heavier burden [Li and Vuong 2005]. To preserve the battery life, mobile nodes should not belong to the common DHT-based P2P overlay network. Scalability of a horizontal P2P system needs to be examined at two levels: (1) at the level of the entire system, the scalability is very good since adding a new group inflicts only a minor additional load on the common DHT-based P2P network; (2) at the level of the groups, scalability is again dependent on the type of P2P network used. Security is inherently very good, because the group-related messaging is restricted to the members of a group, all of which reside in the same autonomous P2P overlay network [Li and Vuong 2005].

4. DISCUSSION AND CONCLUSIONS

P2P systems are ideal platforms for distributed computing and interpersonal communication, which emphasizes the importance of groups and their efficient management. This article presents a conceptual analysis of P2P group management systems. We describe how groups are formed using different P2P system architectures, and analyze the advantages and disadvantages of each P2P system architecture for implementing P2P group management. The evaluation criteria used in the analysis are performance, robustness, fairness, suitability for battery-powered devices, scalability, and security. These properties were selected because they constitute an essential set of challenges faced by the P2P group management systems.

Based on the knowledge provided in the article, the task of designing a P2P group management system can be facilitated by progressing through the following six steps: First, clearly define the purpose why groups are needed. In order to provide assistance for the task, this article provides an extensive sample of existing P2P group management systems found in the literature. By studying these systems, information about alternative approaches to achieve similar kinds of aims can be discovered. Second, examine the presented set of P2P group management procedures to gain an understanding what kinds of procedures need to be implemented, and whether or not some procedures need to be implemented at all. In relation to group management procedures, it is also important to consider whether the groups can be overlapping, whether inter-group messaging is needed, whether the group members need to be aware of each other, and whether the proximity information is taken advantage of in the group establishment. As a third step, estimate the characteristics of the intended use environment. This includes considerations, for instance, on the predicted size of the P2P overlay network, amount of churn, heterogeneity of nodes, and average size of a group. Fourth, define the required properties of the underlying P2P system architecture. By the required properties, we mean the following: performance, robustness, fairness, suitability for battery-powered devices, scalability, and security.

Based on the information gathered in the previous steps, fifth consider how to implement the needed procedures. It is important to notice that in many cases, the underlying P2P system architecture at the same time guides and restricts how each P2P group management procedure can be implemented in an efficient manner. Finally, select an underlying P2P system architecture for implementing the defined P2P group management system. Our conceptual analysis provides a good foundation for finding the appropriate candidates from the diversity of existing P2P system architectures.

According to our survey, a majority of the existing P2P group management systems rely on single-overlay P2P system architectures, even though multioverlay P2P architectures provide many useful features for implementing groups in a P2P system. The fact that vertical P2P systems exploit routing-related information when constructing the multilayered structure makes them ideal for groups that are, for instance, established based on geographical areas. Especially the vertical P2P systems have good performance, fairness, and scalability, but unfortunately robustness and suitability for battery-powered devices can be significantly weakened by the complexity of the multilayered structure. For instance, a mobile node might need to participate in the maintenance of multiple DHT-based overlay networks. The most important advantage of horizontal P2P systems is the complete autonomy of groups that can be implemented using any type of P2P systems, chosen according to the group-specific needs. If the used types of P2P systems can be selected in a near-optimal manner, horizontal P2P systems provide a very interesting alternative for implementing multipurpose P2P group management systems.

In future research, it would be beneficial to combine the systems perspective, adopted in this article, with the social dimensions of P2P group management. This is because the features of a social application often dictate what is needed from a group management system in a technical sense.

ACKNOWLEDGMENTS

The authors would like to acknowledge the anonymous reviewers for their valuable comments throughout the review process.

REFERENCES

AGOSTINI, A. AND MORO, G. 2004. Identification of communities of peers by trust and reputation. In *Proceedings of the 11th International Conference on Artificial Intelligence: Methodology, Systems, and Applications (AIMSA'04)*. Lecture Notes in Computer Science, vol. 3192, Springer, 85–95.

AHMED, R. AND BOUTABA, R. 2011. A survey of distributed search techniques in large scale distributed systems. *IEEE Comm. Surv. Tutorials 13*, 2, 150–167.

ANDROUTSELLIS-THEOTOKIS, S. AND SPINELLIS, D. 2004. A survey of peer-to-peer content distribution technologies. *ACM Comput. Surv. 36,* 4, 335–371.

ANGLADE, A., TIEMANN, M., AND VIGNOLI, F. 2007a. Complex-Network theoretic clustering for identifying groups of similar listeners in p2p systems. In *Proceedings of the ACM Conference on Recommender Systems*. 1–48.

ANGLADE, A., TIEMANN, M., AND VIGNOLI, F. 2007b. Virtual communities for creating shared music channels. In *Proceedings of the 8th International Conference on Music Information*. 1–6.

ARNEDO-MORENO, J. AND HERRERA-JOANCOMARTÍ, J. 2008. Collaborative groupmembership and access control for jxta. In *Proceedings of the 3rd International Conference on Communication System Software and Middleware*. 159–166.

ARTIGAS, M., LÓPEZ, P., AND SKARMETA, A. 2005. Cyclone: A novel design schema for hierarchical dhts. In *Proceedings of the 5th IEEE International Conference on Peer-to-Peer Computing*. 49–56.

BANERJEE, S., BHATTACHARJEE, B., AND KOMMAREDDY, C. 2002. Scalable application layer multicast. In *Proceedings of the Annual Conference of the Special Interest Group on Data Communication*. 205–217.

BHARAMBE, A. R., AGRAWAL, M., AND SESHAN, S. 2004. Mercury: Supporting scalable multi-attribute range queries. In *Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*. 353–366.

BISWAS, D. AND VIDYASANKAR, K. 2008. A highly flexible data structure for multi-level visibility of p2p communities. In *Proceedings of the 9th International Conference on Distributed Computing and Networking (ICDCN '08)*. Lecture Notes in Computer Science, vol. 4904, Springer, 363–374.

BITTORRENT. 2011. http://www.bittorrent.com.

BLOEHDORN, S., HAASE, P., HEFKE, M., SURE, Y., AND TEMPICH, C. 2005. Intelligent community lifecycle support. In *Proceedings of the 5th International Conference on Knowledge Management*. 1–8.

BOTTAZZI, D., MONTANARI, R., AND ROSSI, G. 2008. A self-organizing group management middleware for mobile ad-hoc networks. *Comput. Comm. 31,* 3040–3048.

BRANDS, E. H. T. B, AND KARAGIANNIS, G. 2009. Taxonomy of p2p applications. In *Proceedings of the 3rd IEEE Workshop on Enabling Service-Oriented Internet, GLOBECOM Workshops.* 1–8.

BUYUKKAYA, E., ABDALLAH, M., CAVAGNA, R., AND HU, S. Y. 2008. GROUP: Dual-Overlay state management for p2p nve. In *Proceedings of the 14th IEEE International Conference on Parallel and Distributed Systems*. 817–822.

CASTANO, S. AND MONTANELLI, S. 2005. Semantic self-formation of communities of peers. In *Proceedings of the ESWC Workshop on Ontologies in Peer-to-Peer Communities.* 1–12.

CASTRO, M., DRUSCHEL, P., KERMARREC, A.-M., AND ROWSTRON, A. 2002. SCRIBE: A large-scale and decentralized application-level multicast infrastructure. *IEEE J. Select. Areas Comm. 20,* 8, 100–110.

CASTRO, M., DRUSCHEL, P., KERMARREC, A.-M., NANDI, A., ROWSTRON, A., AND SINGH, A. 2003. SplitStream: High-Bandwidth multicast in cooperative environments. In *Proceedings of the 19th ACM Symposium on Operating Systems Principles.* 298–313.

CHAWATHE, Y., MCCANNE, S., AND BREWER, E. A. 2000. RMX: Reliable multicast for heterogeneous networks. In *Proceedings of the 19th Joint Conference of the IEEE Computer and Communication Societies*. 795–804.

DAS, T., NANDI, S., AND GANGULY, N. 2009. Community formation and search in p2p: A robust and self-adjusting algorithm. In *Proceedings of the 3rd Workshop on Intelligent Networks: Adaptation, Communication and Reconfiguration.* 1–8.

DELMASTRO, F. AND PASSARELLA, A. 2005. An experimental study of p2p group-communication applications in real-world MANETs. In *Proceedings of the IEEE ICPS Workshop on Multi-Hop Ad Hoc Networks: From Theory to Reality (REALMAN'05)*. 1–8.

FAN, B., LUI, J. C. S., AND CHIU, D.-M. 2009. The design trade-offs of bit-torrent-like file sharing protocols. *IEEE/ACM Trans. Network. 17,* 2, 365–376.

FREEDMAN, M. AND MAZIÈRES, D. 2003. Sloppy hashing and self-organizing clusters. In *Peer-to-Peer Systems II*. Lecture Notes in Computer Science, vol. 2735, Springer, 45–55.

FREENET 2011. http://freenetproject.org.

GALATOPOULLOS, D. G., KALOFONOS, D. N., AND MANOLAKOS, E. S. 2008. A p2p soa enabling group collaboration through service composition. In *Proceedings of the International Conference on Pervasive Services.* 111–120.

GANESAN, P., GUMMADI, K., AND GARCIA-MOLINA, H. 2004. Canon in G major: Designing DHTs with hierarchical structure. In *Proceedings of the 24th International Conference on Distributed Computing Systems (ICDCS'04)*. 263–272.

GARCÉS-ERICE, L., BIERSACK, E. W., FELBER, P. A., ROSS, K. W., AND URVOY-KELLER, G. 2003. Hierachical peer-to-peer systems. In *Proceedings of the ACM/IFIP International Conference on Parallel and Distributed Computing.* 643–657.

GHEORGHE, G., LOCIGNO, R., AND MONTRESOR, A. 2011. Security and privacy issues in p2p streaming systems: A survey. *Peer-to-Peer Netw. Appl. 4,* 2, 75–91.

GONG, LI. 2001. JXTA: A network programming environment. *IEEE Internet Comput. 5,* 3, 88–95.

GU, W. AND WEI, W. 2006. Automatic community discovery in peer-to-peer systems. In *Proceedings of the 5th International Conference on Grid and Cooperative Computing Workshops.* 110–116.

GUPTA, S. AND BOSTROM, R. P. 2005. Theoretical model for investigating the impact of knowledge portals on different levels of knowledge processing. *Int. J. Knowl. Learn. 1,* 4, 287–304.

GUPTA, A. AND MOHAPATRA, P. 2007. Energy consumption and conservation in wifi based phones: A measurement-based study. In *Proceedings of the Conference on Sensor, Mesh and Ad Hoc Communications and Networks.* 122–131.

HALEPOVIC, E. AND DETERS, R. 2003. The costs using JXTA. In *Proceedings of the 3rd International Conference on Peer-to-Peer Computing (P2P'03)*.

HARJULA, E., YLIANTTILA, M., ALA-KURIKKA, J., RIEKKI, J., AND SAUVOLA, J. 2004. Plug-And-Play application platform: Towards mobile peer-to-peer. In *Proceedings of the 3rd International Conference on Mobile and Ubiquitous Multimedia.* 63–69.

HARVEY, N., JONES, M., SAROIU, S., THEIMER, M., AND WOLMAN, A. 2003. SkipNet: A scalable overlay network with practical locality properties. In *Proceedings of the 4th USENIX Symposium on Internet Technologies and Systems.* 1–14.

HAUTAKORPI, J. AND CAMARILLO, G. 2007. Evaluation of dhts from the viewpoint of interpersonal communications. In *Proceedings of the International Conference on Mobile and Ubiquitous Multimedia.* 74–83.

HAVERINEN, H., SIREN, J., AND ERONEN, P. 2007. Energy consumption of always-on applications in wcdma networks. In *Proceedings of the 65th Semi-Annual IEEE Vehicular Technology Conference.* 964–968.

JENNINGS, C., LOWEKAMP, B., RESCORLA, E., BASET, S., AND SCHULZRINNE, H. 2010. Resource location and discovery (RELOAD) base protocol. IETF: draft-ietf-p2psip-base-12.

KELÉNYI, I. AND NURMINEN, J. K. 2008. Optimizing energy consumption of mobile nodes in heterogeneous kademlia-based distributed hash tables. In *Proceedings of the International Conference on Next Generation Mobile Applications, Services, and Technologies.* 70–75.

KHAMBATTI, M., RYU, K., AND DASGUPTA, P. 2002a. Efficient discovery of implicitly formed peer-to-peer communities. *Int. J. Parallel Distrib. Syst. Netw. 5,* 4, 155–164.

KHAMBATTI, M., RYU, K., AND DASGUPTA, P. 2002b. Peer-to-Peer communities: Formation and discovery. In *Proceedings of the International Conference on Parallel and Distributed Computing Systems.* 161–166.

KHAMBATTI, M., RYU, K., AND DASGUPTA, P. 2004. Structuring peer-to-peer networks using interest-based communities. In *Proceedings of the International Workshop on Databases, Information Systems, and Peer-to-Peer Computing*. Lecture Notes in Computer Science, vol. 2944, Springer, 48–63.

KOBAYASHI, H., TAKIKAWA, H., INABA, T., AND TAKIKAWA, Y. 2005. A self-organizing overlay network to exploit the locality of interests for effective resource discovery in p2p systems. In *Proceedings of the Symposium on Applications and the Internet.* 246–255.

KOSKELA, T., KASSINEN, O., OU, Z., AND YLIANTTILA, M. 2010. Improving community management performance with two-level hierarchical dht overlays.*J. Internet Technol. 11,* 2, 167–181.

KRISHNAMURTHY, B., WANG, J., AND XIE, Y. 2001. Early measurement of a cluster-based architecture for p2p systems. In *Proceedings of the ACM SIGCOMM Internet Measurement Workshop.* 105–109.

KWOK, S. H., AND GAO, S. 2003. Knowledge sharing community in p2p network and its application features: A study of motivation perspective. In *Proceedings of the 7th Pacific Asia Conference on Information Systems.* 1539–1549.

LEI, J., AND FU, X. 2009. Interest-Based peer-to-peer group management. In *Proceedings of the 2nd International Workshop on Future Multimedia Networking.* Lecture Notes in Computer Science, vol. 5630, Springer, 107–118.

LEONARD, D., YAO, Z., RAI, V., AND LOGUINOV, D. 2007. On lifetime-based node failure and stochasticresilience of decentralized peer-to-peer networks. *IEEE/ACM Trans. Network. 15,* 3, 644–656.

LEONARD, D., YAO, Z., WANG, X., AND LOGUINOV, D. 2008. On static and dynamic partitioning behavior of large-scale p2p networks. *IEEE/ACM Trans. Network. 16,* 6, 1475–1488.

LI, J., STRIBLING, J., AND GIL, T. M. 2004. Comparing the performance of distributed hash tables under churn. In *Proceedings of the 3rd International Workshop on Peer-to-Peer Systems*. Lecture Notes in Computer Science, vol. 3279, Springer, 87–99.

LI, J. AND VUONG, S. 2006. Grid resource discovery based on semantic p2pP communities. In *Proceedings of the 21st ACM Symposium on Applied Computing.* 754–758.

LI, J., AND VUONG, S. 2005. Ontology-Based clustering and routing in peer-to-peer networks. In *Proceedings of the 6th International Conference on Parallel and Distributed Computing, Applications and Technologies.* 791–795.

LI, J., YU, X., WU, X., WANG, R., AND SUN, J. 2007. Semantic community in a peer-to-peer network.*Adv. Soft Comput. Advances Intell. Web Master. 43,* 192–197.

LI, Y. -M., TAN, Y., AND ZHOU, Y. -P. 2008. Analysis of scale effects in peer-to-peer networks. IEEE/ACM Trans. Network. *16*, 3, 590–602.

LI, Z., DONG, Y., ZHUANG, L., AND HUANG, J. 2003. Implementation of secure peer group in peer-to-peer network. In *Proceedings of the International Conference on Communication Technology.* 192–195.

LIOTTA, A., BALLETTE, M., LIN, L., GASPARONI, M., BRICK, P., AND PAPADOGLOU, N. 2005. Service-Driven group management for mobile p2p services. *IFIP- Int. Federation Inf. Process. Intell. Comm. Syst. 190,* 221–230.

LIU, K., BHADURI, K., DAS, K., NGUYEN, P., AND KARGUPTA, H. 2006. Client-Side web mining for community formation in peer-to-peer environments. *ACM SIGKDD Explor. 8,* 2, 11–20.

LU, J., AND CALLAN, J. 2003. Content-Based retrieval in hybrid peer-to-peer networks. In *Proceedings of ACM Conference on Information and Knowledge Management.* 199–206.

MAYMOUNKOV, P. AND MAZIERES D. 2002. Kademlia: A peer-to-peer information system based on the xor metric. In *Proceedings of the International Workshop on Peer-to-Peer Systems.* 1–6.

MEI, H. AND CHANG, S. 2004. PP-COSE: A p2p community search scheme. In *Proceedings of the 4th International Conference on Computer and Information Technology.* 416–423.

MONTANELLI, S. 2006. Emergent communities for semantic collaboration in multi-knowledge environments: Methods and techniques. In *Proceedings of the 13th Doctoral Consortium on Advanced Information Systems Engineering.* 1152–1159.

MORÁVEK, A. AND JELÍNEK, I. 2004. Using centralized element in p2p network for better community management. In *Proceedings of the International Conference on Computer Systems and Technologies.* 1–6.

NAPSTER 2011. http://free.napster.com.

NURMINEN, J. K. AND NÖYRÄNEN, J. 2008. Energy-Consumption in mobile peer-to-peer – quantitative results from file sharing sharing. In *Proceedings of the 5th IEEE Consumer Communications and Networking Conference.* 729–733.

OU, Z., HARJULA, E., KOSKELA, T., AND YLIANTTILA, M. 2010. GTPP: General truncated pyramid peer-to-peer architecture over structured dht networks. *ACM Mobile Network. Appl. 15,* 729–749.

PALOMAR, E., ESTEVEZ-TAPIADOR, J. M., HERNANDEZ-CASTRO, J. C., AND RIBAGORDA, A. 2006. Security in p2p networks: Survey and research directions. In *Proceedings of the Workshop on Emerging Directions in Embedded and Ubiquitous Computing (EUC).* Lecture Notes in Computer Science, vol. 4097, Springer, 183–192.

RAO, A., LAKSHMINARAYANAN, K., SURANA, S., KARP, R., AND STOICA, I. 2003. Load balancing in structured p2p systems. In *Proceedings of the 2nd International Workshop on Peer-to Peer Systems II*. Lecture Notes in Computer Science, vol. 2735, Springer, 68–79.

RATNASAMY, S., FRANCIS, P., HANDLEY, M., KARP, R., AND SHENKER, S. 2001. A scalable content-addressable network. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications.* 161–172.

RATNASAMY, S., HANDLEY, M., KARP, R., AND SHENKER, S. 2002. Topologically-aware overlay construction and server selection. In *Proceedings of the 21st Joint Conference of the IEEE Computer and Communications Societies.* 1190–1199.

RAVI, N., SCOTT, J., HAN, L., AND IFTODE, L. 2008. Context-Aware battery management for mobile phones. In *Proceedings of the 6th IEEE Conference on Pervasive Computing and Communications.* 224–233.

RAVICHANDRAN, A. AND YOON, J. 2006. Trust management with delegation in grouped peer-to-peer communities. In *Proceedings of the 11th ACM Symposium on Access Control Models and Technologies.* 71–80.

RHEINGOLD, H. 2000. The Virtual Community: Homesteading on the Electronic Frontier. MIT Press, Cambrdge, MA.

RIPEANU, M. 2001. Peer-to-Peer architecture case study: Gnutella network. In *Proceedings of the International Conference on Peer-to-Peer Computing.* 99–100.

RISSON, J. AND MOORS, T. 2006. Survey of research towards robust peer-to-peer networks: Search methods. *Comput. Network. 50,* 17, 3485–3521.

RODRIGUES, R. AND DRUSCHEL, P. 2010. Peer-to-Peer systems. *Comm. ACM 53,* 10, 72–82.

ROWSTRON, A. AND DRUSCHEL, P. 2001 Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In *Proceedings of the IFIP/ACM International Conference on Distributed Systems Platforms.* 329–350.

SAKARYAN, G., UNGER, H., AND LECHNER, U. 2004. About the value of virtual communities in p2p networks. In *Proceedings of the 4th IEEE International Symposium and School on Advance Distributed Systems.* 170–185.

SHAN, X. AND SHRIRAM, A. 2006. Enterprise mobile applications based on presence and logical proximity. In *Proceedings of the International Conference on Wireless Communications and Mobile Computing.* 683–688.

SKYPE 2011. http://www.skype.com.

SRIPANIDKULCHAI, K., MAGGS, B., AND ZHANG, H. 2003. Efficient content location using interest-based locality in peer-to-peer systems. In *Proceedings of the 22nd Joint Conference of the IEEE Computer and Communications Societies.* 2166–2176.

STOICA, I., MORRIS, R., KARGER, D., KAASHOEK, M., AND BALAKRISHNAN, H. 2001. Chord: A scalable peer-to-peer lookup service for internet applications. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications.* 149–160.

SUN, Y., SUN, L., HUANG, X., AND LIN, Y. 2006. Resource discovery in locality-aware group-based semantic overlay of peer-to-peer networks. In *Proceedings of the 1st ACM International Conference on Scalable Information Systems.*

THAM, J.-Y., LEE, S.-L., TAN, C.-E., AND TEE, L.-C. 2004. A distributed peer-to-peer platform for synchronized group collaboration and knowledge sharing. In *Proceedings of International Symposium on Distributed Computing and Applications to Business, Engineering and Science.* 94–99.

VALDURIEZ, P. AND PACITTI, E. 2004. Data management in large-scale p2p systems. In *Proceedings of the Conference on Vector and Parallel Processing.* 104–118.

WANG, Y. AND VASSILEVA, J. 2004. Trust-Based community formation in peer-to-peer file sharing networks. In *Proceedings of the International Conference on Web Intelligence.* 341–348.

XU, Z., MIN, R., AND HU, Y. 2003. HIERAS: A DHT based hierarchical p2p routing algorithm. In *Proceedings of the International Conference on Parallel Processing.* 187–194.

XUE, G. -T., YOU, J. -Y., AND JIA, Z. -Q. 2004. An interest group model for content location in peer-to-peer systems. In *Proceedings of the IEEE International Conference on E-Commerce Technology for Dynamic E-Business.* 306–309.

YAMAMOTO, A., ARAKI, Y., AND SWEENEY, M. 2008. A framework of community management with object deputy mechanism for IP multimedia subsystem. In *Proceedings of the International Symposium on Applications and the Internet.* 463–466.

YANG, B., AND GARCIA-MOLINA, H. 2003. Designing a super-peer network. In *Proceedings of the 19th International Conference on Data Engineering.* 49–60.

YAO, Z., WANG, X., LEONARD, D., AND LOGUINOV, D. 2009. Node isolation model and age-based neighbor selection in unstructured p2p networks. *IEEE/ACM Trans. Network. 17,* 1, 144–157.

YU, J. AND LI, M. 2008. CBT: A proximity-aware peer clustering system in large-scale bittorrent-like peer-to-peer networks. *Comput. Comm. 31,* 591–602.

ZHANG, J., LIU, L., RAMASWAMY, L., ZHANG, G., AND PU, C. 2007. A utility-aware middleware architecture for decentralized group communication applications. In *Proceedings of the ACM/IFIP/USENIX International Conference on Middleware*. Lecture Notes in Computer Science, vol. 4834, Springer, 410–428.

ZHAO, B., DUAN, Y., HUANG, L., JOSEPH, A., AND KUBIATOWICZ, J. 2002. Brocade: Landmark routing on overlay networks. In *Proceedings of the 1st International Workshop on Peer-to-Peer Systems (Revised Papers)*. Lecture Notes in Computer Science, vol. 2429, Springer, 34–44.

ZHAO, B., HUANG, L., STRIBLING, J., RHEA, S., JOSEPH, A., AND KUBIATOWICZ, J. 2004. Tapestry: A resilient global-scale overlay for service deployment. *IEEE J. Selected Areas Comm. 22,* 1, 41–53.

ZOELS, S., DESPOTOVIC, Z., AND KELLERER, W. 2008. On hierarchical dht systems–an analytical approach for optimal designs. *Comput. Comm. 31,* 576–590.

ZHUANG, S. Q., ZHAO, B. Y., JOSEPH, A. D., KATZ, R. H., AND KUBIATOWICZ, J. D. 2001. Bayeux: An architecture for scalable and fault-tolerant wide-area data dissemination. In *Proceedings of the 11th International Workshop on Network and Operating Systems Support for Digital Audio and Video.* 11–20.

Received August 2010; revised September 2011; accepted October 2011